

## Article

# Synchronization Control for a Mobile Manipulator Robot (MMR) System: A First Approach Using Trajectory Tracking Master–Slave Configuration

Jorge Gustavo Pérez-Fuentevilla <sup>1</sup>, América Berenice Morales-Díaz <sup>2,\*</sup> and Alejandro Rodríguez-Ángeles <sup>1,†</sup>

<sup>1</sup> Departamento de Ingeniería Eléctrica, CINVESTAV-IPN, Mexico City 07360, Mexico; jorge.perezf@cinvestav.mx (J.G.P.-F.); aangeles@cinvestav.mx (A.R.-Á.)

<sup>2</sup> Robótica y Manufactura Avanzada, CINVESTAV-Saltillo, Ramos Arizpe 25900, Mexico

\* Correspondence: america.morales@cinvestav.edu.mx

† These authors contributed equally to this work.

**Abstract:** In cooperative tasks, the ability to keep a kinematic relationship between the robots involved is essential. The main goal in this work is to design a synchronization control law for mobile manipulator robots (MMRs) considering a (2,0) differential mobile platform, which possesses a non-holonomic motion constraint. To fulfill this purpose, a generalized trajectory tracking control law based on the computed torque technique, for an MMR with  $n$  degrees of freedom, is presented. Using Lyapunov stability theory, it is shown that the closed loop system is semiglobal and uniformly ultimately boundedness (UUB) stable. To add position-level static coupling terms to achieve synchronization on a group of MMRs, the control law designed for the trajectory tracking problem is extended. Both experimental and numerical simulation results are presented to show the designed controllers performance. A successful experimental validation for the trajectory tracking problem using an 8 degrees of freedom (DoF) robot model (KUKA youBot) is depicted. Finally, numerical simulations in the CoppeliaSim environment are shown, which are used to test the synchronization control law made on the hypothetical scenario, where a two robot system has to manipulate an object over a parametric trajectory.



**Citation:** Pérez-Fuentevilla, J.G.; Morales-Díaz, A.B.; Rodríguez-Ángeles, A. Synchronization Control for a Mobile Manipulator Robot (MMR) System: A First Approach Using Trajectory Tracking Master–Slave Configuration. *Machines* **2023**, *11*, 962. <https://doi.org/10.3390/machines11100962>

Academic Editor: Dan Zhang

Received: 12 July 2023

Revised: 7 September 2023

Accepted: 7 September 2023

Published: 16 October 2023



**Copyright:** © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/> 4.0/).

## 1. Introduction

Entrusting a single robot to deal with complex or heavy tasks may not be the best option: some tasks require more than one robot, so in that sense, the synchronization of multi-robot systems represents a better solution. For this reason, it has been an intensive area of research in the last decades. Cooperative manipulation is one of the methods used to deal with the synchronization idea; it allows a multi-robot system to manipulate a heavy and/or big object more efficiently than a single robot.

There are some classical approaches to achieve coordination on a robot system [1]: one of them is the master–slave strategy, which is used in this work. In this strategy, a master MMR is considered, which is in charge of setting up the trajectory reference for the slave MMRs while using cross-coupled controls to synchronize motions amongst the robots; therefore, a kinematic relationship can be maintained for a desired task.

There are some interesting approaches for synchronizing fixed-base manipulators, for example, synchronization control based on output differential flatness [2]; in [3], the robustness of the sliding mode technique is considered; in [4], a system under parametric uncertainties is considered; in [5], the authors deal with dynamical uncertainties; and in [6], the problem of obstacle avoidance is solved. More recently, sophisticated controllers have

been considered, see, for instance, [7], where a non-linear model predictive control is used for the synchronization problem at position and force levels.

Mobile robots are also a case of study on synchronization. In [1], a swarm of mobile robots that can switch between formations are presented; in [8], a discrete approach on differential robots is developed; also, varying formations have been considered; this is the case in [9].

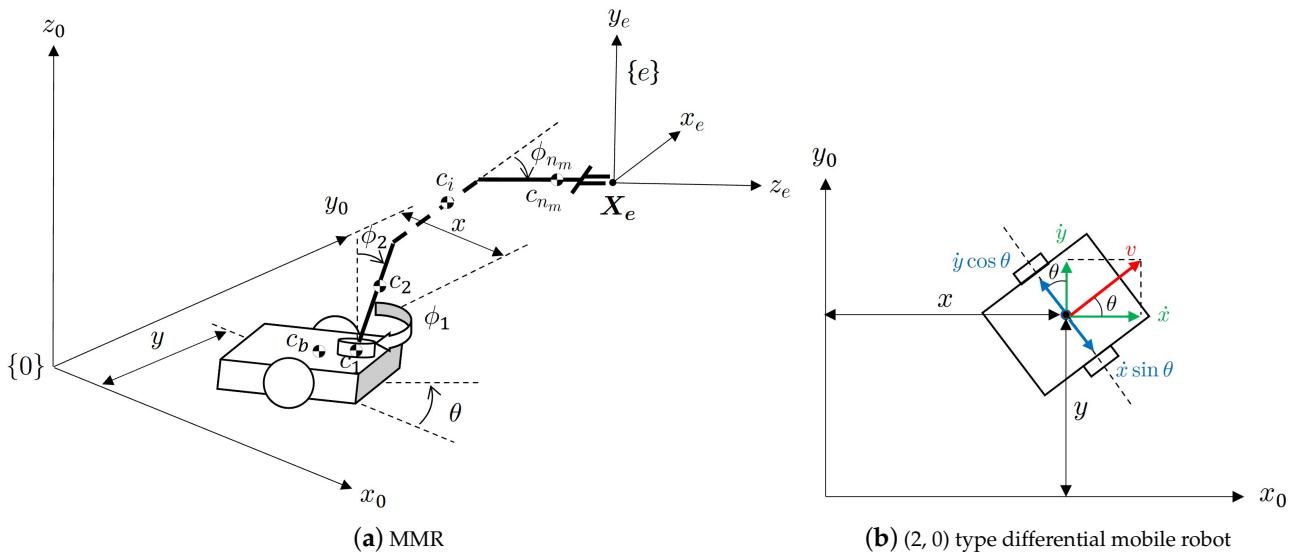
For the case of control MMRs, there are some works reported for the trajectory tracking control; in [10], the authors solved the trajectory tracking problem by designing a robust controller based-on sliding mode technique; an extended state observer and movement constraint in the mobile base was introduced in [11]; and another approach was taken in [12], where the dynamical model of MMR is obtained by using Gibbs–Appell formulation instead of the classical Lagrange methodology. However, these works do not consider the synchronization approach.

The present work has the goal of synchronizing MMRs; to this end, a position-level synchronization control law using a master–slave scheme with cross-coupled terms is provided as an immediate extension of a trajectory tracking controller. The trajectory tracking problem for a  $n$  DoF MMR is solved designing a non-linear control law based-on a computed torque control with gravity compensation.

This work is organized as follows. In Section 2, the dynamic equations of the  $n$  DoF MMR are developed. In Section 3, the controller design and stability analysis are addressed. The master–slave synchronization is posted as an extension of the trajectory tracking controller in Section 4. The results are depicted in Section 5, and the findings and conclusions are summarized in Section 2.

## 2. Problem Statement

Consider the  $n$  DoF MMR configuration shown in Figure 1, where  $\{0\}$  frame corresponds to inertial frame, and the end-effector frame is denoted by  $\{e\}$ .



**Figure 1.**  $n$  DoF MMR configuration.

Let us define  $n = n_b + n_m$ , where  $n_b = 3$  is the number of DoF for the mobile base and  $n_m$  corresponds to the number of DoF for the manipulator. This way, generalized coordinates are defined as  $\mathbf{q}_b = [x \ y \ \theta]^\top$  for the mobile and  $\mathbf{q}_m = [\phi_1 \ \phi_2 \ \dots \ \phi_{n_m}]^\top$  for the manipulator. This yields  $\mathbf{q}(t) = [\mathbf{q}_b^\top \ \mathbf{q}_m^\top]^\top$  for the whole system, namely,

$$\mathbf{q}(t) = [q_1 \ q_2 \ \dots \ q_n]^\top = [x \ y \ \theta \ \phi_1 \ \phi_2 \ \dots \ \phi_{n_m}]^\top. \quad (1)$$

where  $(x, y, \theta)$  is the Cartesian position and orientation of the mobile base on the  $(x_0, y_0)$  plane and  $\phi_i, i = 1, \dots, n_m$  are the angular positions for the  $i$ -th link.

A  $(2, 0)$  type differential mobile robot is considered, meaning that the MMR platform is governed by the kinematic model shown in (2) for the mobile part.

$$\begin{cases} \dot{x} &= v \cos \theta \\ \dot{y} &= v \sin \theta \\ \dot{\theta} &= \omega. \end{cases} \quad (2)$$

where  $(\dot{x}, \dot{y})$  are the Cartesian velocities,  $\dot{\theta}$  is its angular velocity on the  $(x_0, y_0)$  plane, and  $v$  is the longitudinal velocity. From (2), it is easy to obtain the non-holonomic constraint that the mobile robot satisfies as

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0. \quad (3)$$

### 2.1. Forward and Inverse Kinematics

The deal with forward kinematics is being able to describe the pose of the end-effector  $\mathbf{X}_e$  as a function of  $\mathbf{q}(t)$ , i.e.,

$$\mathbf{X}_e = \begin{bmatrix} {}^0\mathbf{P}_e \\ {}^0\mathbf{O}_e \end{bmatrix} = f(\mathbf{q}), \quad (4)$$

where  ${}^0\mathbf{P}_e$  is the position vector of the end-effector measured from inertial frame  $\{0\}$  and  ${}^0\mathbf{O}_e$  describes its orientation.  $\mathbf{X}_e \in \mathbb{R}^\rho$  is defined on the task-space.

On the other hand, inverse kinematics is about finding solutions for  $\mathbf{q}(t)$  given a pose vector  $\mathbf{X}_e$ , i.e.,

$$\mathbf{q}(t) = f^{-1}(\mathbf{X}_e). \quad (5)$$

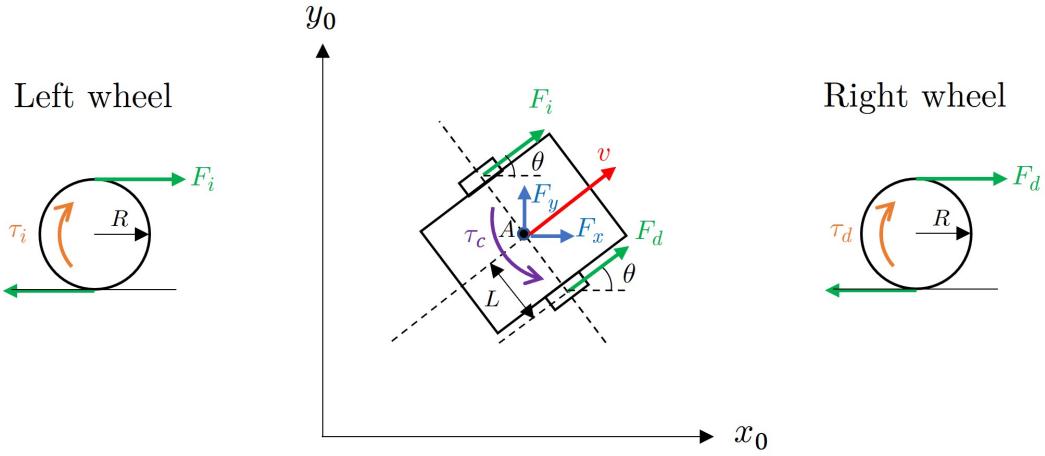
For redundant robots, such as the MMRs considered, solving the inverse kinematics is a complicated task since the number of DoF is greater than the task-space dimension, i.e.,  $n > \rho$ . This implies having multiple solutions for  $\mathbf{q}(t)$  given a desired trajectory, which could be a problem because some discontinuities could appear from one solution to another, but it is also advantageous since the robot can satisfy a task in more than one way.

### 2.2. Generalized Dynamical Model

For a robot without kinematic constraints, the dynamical model can be described as

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{F}_f(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{G}(\mathbf{q}) = \tau - \mathbf{J}_e^\top(\mathbf{q})\mathbf{h}_r. \quad (6)$$

where  $\mathbf{D}(\mathbf{q}) \in \mathbb{R}^{n \times n}$  is the inertia matrix;  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \in \mathbb{R}^{n \times n}$  is the Coriolis matrix;  $\mathbf{F}_f(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$  is the force vector due to friction;  $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^n$  is the gravitational effects vector;  $\mathbf{J}_e(\mathbf{q}) \in \mathbb{R}^{\rho \times n}$  is the Jacobian matrix that relates the articular velocities with the end-effector velocities ( $\dot{\mathbf{X}}_e = \mathbf{J}_e(\mathbf{q})\dot{\mathbf{q}}$ );  $\mathbf{h}_r \in \mathbb{R}^\rho$  is the generalized forces vector due to robot interaction with the environment, if there is no interaction with the environment  $\mathbf{J}_e^\top(\mathbf{q})\mathbf{h}_r = 0$ ;  $\tau = [F_x \ F_y \ \tau_c \ \tau_1 \ \tau_2 \ \dots \ \tau_{n_m}]^\top \in \mathbb{R}^n$  is the control forces vector;  $F_x$  and  $F_y$  are the translation forces on the  $x_0$  and  $y_0$  axis, respectively;  $\tau_c$  is the mobile torque with respect to its rotation axis; and  $\tau_i, i = 1, 2, \dots, n_m$  is the torque of the  $i$ -th link. All the torques and forces on the mobile platform are shown in Figure 2.



**Figure 2.** Forces and torques description on the mobile platform.

The problem with model (6) is that the non-holonomic constraint (3) is not satisfied. The incorporation of the constraint into the dynamical model is achieved through the next procedure, [13,14].

The non-holonomic constraint (3) can be re-written as  $\mathbf{A}(\mathbf{q})\dot{\mathbf{q}} = \mathbf{0}$ , with  $\mathbf{A}(\mathbf{q}) \in \mathbb{R}^{k \times n}$  being  $k = 1$  the number of independent constraints; this yields

$$\mathbf{A}(\mathbf{q}) = [\sin \theta \quad -\cos \theta \quad 0 \quad \mathbf{0}_{1 \times n_m}], \quad (7)$$

where  $\mathbf{0}_{1 \times n_m}$  is a row vector of  $n_m$  zeros.

Let  $\mathbf{S}(\mathbf{q}) \in \mathbb{R}^{n \times (n-k)}$  be an invertible full-rank matrix that spans the null space of  $\mathbf{A}(\mathbf{q})$ , i.e.,  $\mathbf{S}^\top(\mathbf{q})\mathbf{A}^\top(\mathbf{q}) = \mathbf{0}$  is satisfied (or equivalently,  $\mathbf{A}(\mathbf{q})\mathbf{S}(\mathbf{q}) = \mathbf{0}$ ). Thus,

$$\mathbf{S}(\mathbf{q}) = \begin{bmatrix} \cos \theta & 0 & \mathbf{0}_{1 \times n_m} \\ \sin \theta & 0 & \mathbf{0}_{1 \times n_m} \\ 0 & 1 & \mathbf{0}_{1 \times n_m} \\ \mathbf{0}_{n_m \times 1} & \mathbf{0}_{n_m \times 1} & \mathbf{1}_{n_m \times n_m} \end{bmatrix}, \quad (8)$$

with  $\mathbf{0}_{n_m \times 1}$  a column vector of  $n_m$  zeros and  $\mathbf{1}_{n_m \times n_m}$  the identity matrix of dimension  $n_m \times n_m$ .

An auxiliary vector  $\eta(t) \in \mathbb{R}^{n-k}$  exists that satisfies

$$\dot{\mathbf{q}}(t) = \mathbf{S}(\mathbf{q})\eta(t), \quad (9)$$

which is defined as

$$\eta(t) = [\eta_1 \quad \eta_2 \quad \dots \quad \eta_{n-1}]^\top = [v \quad \dot{\theta} \quad \dot{\phi}_1 \quad \dot{\phi}_2 \quad \dots \quad \dot{\phi}_{n_m}]^\top. \quad (10)$$

Let  $\tau_i$  and  $\tau_d$  be the left and right wheels torque, respectively, see Figure 2. It is possible to write  $\tau = \mathbf{B}(\mathbf{q})\tau_{red}$  with  $\mathbf{B}(\mathbf{q}) \in \mathbb{R}^{n \times (n-k)}$ , where:

$$\tau_{red} = [\tau_d \quad \tau_i \quad \tau_1 \quad \tau_2 \quad \dots \quad \tau_{n_m}]^\top \quad (11)$$

$$\mathbf{B}(\mathbf{q}) = \begin{bmatrix} \frac{1}{R} \cos \theta & \frac{1}{R} \cos \theta & \mathbf{0}_{1 \times n_m} \\ \frac{1}{R} \sin \theta & \frac{1}{R} \sin \theta & \mathbf{0}_{1 \times n_m} \\ \frac{L}{R} & -\frac{L}{R} & \mathbf{0}_{1 \times n_m} \\ \mathbf{0}_{n_m \times 1} & \mathbf{0}_{n_m \times 1} & \mathbf{1}_{n_m \times n_m} \end{bmatrix}. \quad (12)$$

With  $R$  being the radius of the wheels and  $L$  the length from the longitudinal axis of the mobile robot to the wheels.

Finally, the constrained model is defined as follows:

$$\begin{aligned}\bar{\mathbf{D}}(\mathbf{q})\dot{\eta} + \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}})\eta + \bar{\mathbf{F}}_f(\mathbf{q}, \dot{\mathbf{q}}) + \bar{\mathbf{G}}(\mathbf{q}) &= \bar{\tau}(\mathbf{q}) \\ \dot{\mathbf{q}} &= \mathbf{S}(\mathbf{q})\eta\end{aligned}. \quad (13)$$

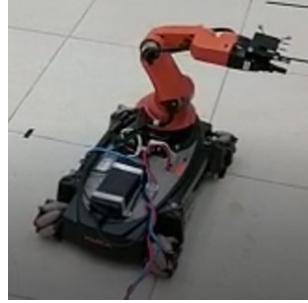
where:

$$\begin{aligned}\bar{\mathbf{D}}(\mathbf{q}) &= \mathbf{S}^T(\mathbf{q})\mathbf{D}(\mathbf{q})\mathbf{S}(\mathbf{q}) \in \mathbb{R}^{(n-k) \times (n-k)} \\ \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) &= \mathbf{S}^T(\mathbf{q})\mathbf{D}(\mathbf{q})\dot{\mathbf{S}}(\mathbf{q}) + \mathbf{S}^T(\mathbf{q})\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{S}(\mathbf{q}) \in \mathbb{R}^{(n-k) \times (n-k)} \\ \bar{\mathbf{F}}_f(\mathbf{q}, \dot{\mathbf{q}}) &= \mathbf{S}^T(\mathbf{q})\mathbf{F}_f(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n-k} \\ \bar{\mathbf{G}}(\mathbf{q}) &= \mathbf{S}^T(\mathbf{q})\mathbf{G}(\mathbf{q}) \in \mathbb{R}^{n-k} \\ \bar{\tau}(\mathbf{q}) &= \mathbf{S}^T(\mathbf{q})\mathbf{B}(\mathbf{q})\tau_{red} - \mathbf{S}^T(\mathbf{q})\mathbf{J}_e^T(\mathbf{q})\mathbf{h}_r \in \mathbb{R}^{n-k}.\end{aligned}$$

$\mathbf{B}(\mathbf{q})$  is called the *motorization matrix*, and it establishes the relationship between the generalized forces vector  $\tau$  of the unconstrained model (6) and a new control vector  $\tau_{red}$  used for the new model (13), where the non-holonomic constraint is satisfied. Notice that the kinematic model (2) is inside the relationship (9); therefore, model (13) satisfies the mobile constraint in a natural way.

### 2.3. Particular Case: KUKA youBot

The KUKA youBot is an 8 DoF industrial-type MMR ( $n = 8$ ) that is also used for research purposes [15]. It is composed of a 5 DoF arm mounted on a omnidirectional mobile base, considered as a differential mobile base in this work by constraining the wheels of each side of the platform so they move at the same velocity. See Figure 3.



**Figure 3.** KUKA youBot.

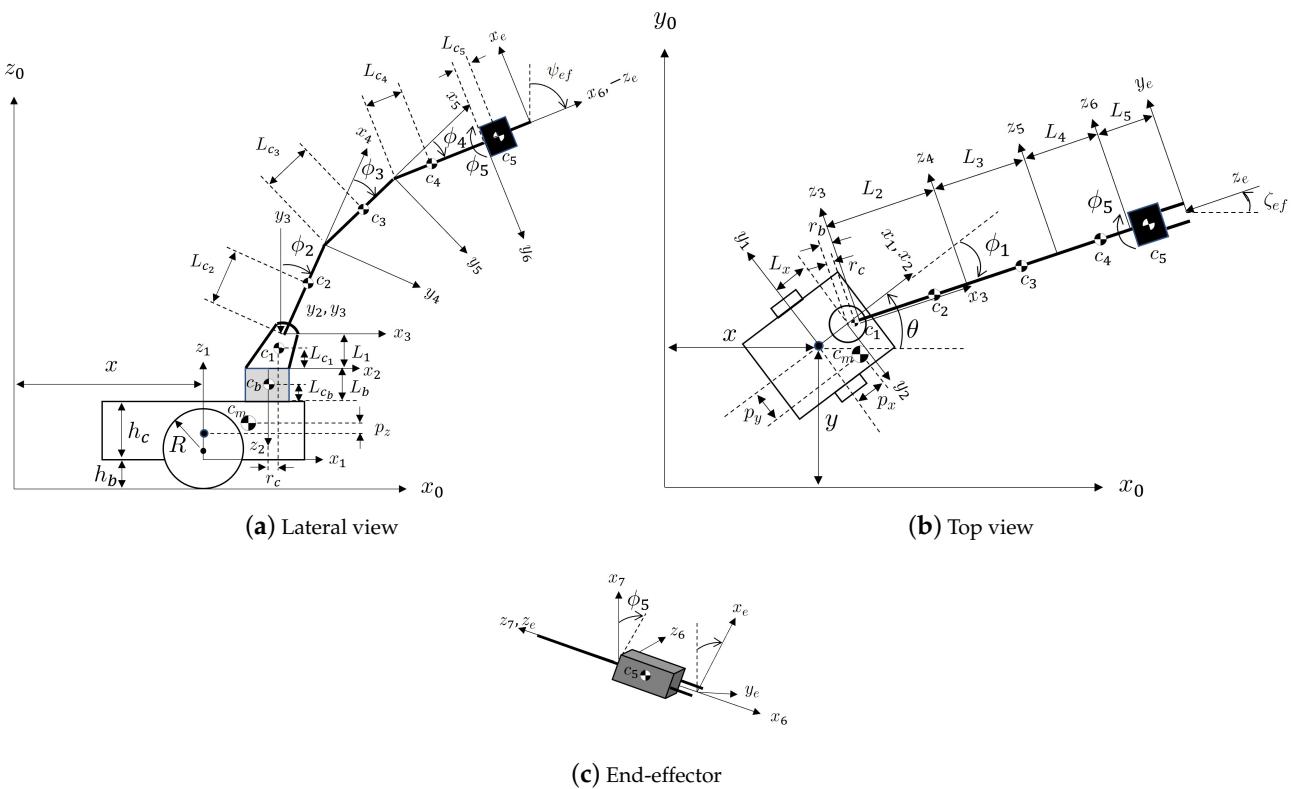
KUKA youBot wheels are of the Mecanum type; these kind of wheels are composed of a series of rollers placed at 45 degrees from the rotation axis and allow for longitudinal and transversal motion; however, when the two wheels of each side rotate at the same angular speed and direction, the transversal velocity is zero, and the omnidirectional base can only have longitudinal and angular velocities (see the kinematic model for an omnidirectional platform in [16]).

The following assumptions for the mobile base are considered:

- The two wheels of each side rotate at the same speed.
- The robot is running on a firm ground surface.
- The four wheels are always in contact with the ground surface.
- There is no slippage between the wheels and the ground surface.

With these considerations, and the fact that the moving direction of KUKA youBot wheels is the mobile longitudinal axis, i.e., it has not orientation movement, the KUKA youBot mobile base can be treated as a (2, 0) type differential mobile base [17].

The MMR configuration is depicted in Figure 4.



**Figure 4.** KUKA youBot configuration.

Consider the symmetrical elements (mobile base and links) in the KUKA configuration, and the inertia tensor matrix of each element is defined as  $\mathbf{I}_i = \text{diag}\{I_{ixx} \quad I_{iyy} \quad I_{izz}\}$ ,  $i = m, b, 1, \dots, 5$  measured in local frame; then, the parameters are defined as follows:

- $c_m, c_b, c_i, i = 1, \dots, 5$ : Mobile base, fixed link, and  $i$ -th link center of mass (CoM).
  - $m_m, m_b, m_i, i = 1, \dots, 5$ : Mobile base, fixed link, and  $i$ -th link mass.
  - $h_b, h_c$ : Distance from the floor to mobile base, and mobile base height, respectively.
  - $R$ : Wheels radius.
  - $p_x, p_y, p_z$ : Distances to the mobile base CoM.
  - $L$ : Distance measured from mobile base longitudinal axis to its wheels.
  - $L_i, i = b, 1, \dots, 5$ :  $i$ -th link length.
  - $L_{ci}, i = b, 1, \dots, 5$ : Length to the  $i$ -th link CoM.
  - $r_b, r_c$ : Radius measured on the  $(x_0, y_0)$  plane, from the fixed link CoM to the second link base, and to the first link CoM, respectively.
  - $L_x$ : Distance measured on the  $(x_0, y_0)$  plane, from the mobile base centroid to the fixed link centroid.
  - ${}^i\mathbf{I}_i, i = m, b, 1, \dots, 5$ :  $i$ -th element inertia tensor.
  - $b_{q_i}, i = 1, \dots, 8$ : viscous friction coefficient related to the generalized coordinate  $q_i$ .

### 2.3.1. Forward Kinematics

Let us define  ${}^0\mathbf{P}_e = [x_{ef} \ y_{ef} \ z_{ef}]^\top$  as the position vector for the end-effector on the inertial frame, and  ${}^0\mathbf{O}_e = [\psi_{ef} \ \zeta_{ef}]^\top$  for the orientation, where  $\psi_{ef}$  is the end-effector orientation with respect to the vertical axis ( $z_0$ ) and  $\zeta_{ef}$  is the end-effector orientation with respect to the horizontal axis ( $x_0$ ); the third orientation for the end-effector would be directly affected by  $\phi_5$ , namely, the wrist rotation; however, for the task contemplated in this work

this is not necessary, and it is not taken into account. The above considerations allow one to define the end-effector pose as follows:

$$\mathbf{X}_e = [x_{ef} \ y_{ef} \ z_{ef} \ \psi_{ef} \ \zeta_{ef}]^\top. \quad (14)$$

The end-effector is related to the inertial frame  $\{0\}$  through the homogeneous matrix defined as:

$${}^0\mathbf{T}_e = \begin{bmatrix} {}^0\mathbf{R}_e & {}^0\mathbf{P}_e \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix}. \quad (15)$$

where:

$${}^0\mathbf{R}_e = \begin{bmatrix} \sin(\phi_1 - \theta) \sin \phi_5 - \cos(\phi_1 - \theta) \alpha_1 \cos \phi_5 & \sin(\phi_1 - \theta) \cos \phi_5 + \alpha_1 \cos(\phi_1 - \theta) \sin \phi_5 & -\alpha_2 \cos(\phi_1 - \theta) \\ \cos(\phi_1 - \theta) \sin \phi_5 + \alpha_1 \sin(\phi_1 - \theta) \cos \phi_5 & \cos(\phi_1 - \theta) \cos \phi_5 - \alpha_1 \sin(\phi_1 - \theta) \sin \phi_5 & \alpha_2 \sin(\phi_1 - \theta) \\ \alpha_2 \cos \phi_5 & -\alpha_2 \sin \phi_5 & -\alpha_1 \end{bmatrix} \quad (16)$$

$${}^0\mathbf{P}_e = \begin{bmatrix} x + L_x \cos \theta + [r_b + L_2 \sin \phi_2 + L_3 \sin(\phi_2 + \phi_3) + (L_4 + L_5) \sin(\phi_2 + \phi_3 + \phi_4)] \cos(\phi_1 - \theta) \\ y + L_x \sin \theta - [r_b + L_2 \sin \phi_2 + L_3 \sin(\phi_2 + \phi_3) + (L_4 + L_5) \sin(\phi_2 + \phi_3 + \phi_4)] \sin(\phi_1 - \theta) \\ h_b + h_c + L_1 + L_b + L_2 \cos \phi_2 + L_3 \cos(\phi_2 + \phi_3) + (L_4 + L_5) \cos(\phi_2 + \phi_3 + \phi_4) \end{bmatrix} \quad (17)$$

$$\alpha_1 = \cos(\phi_2 + \phi_3 + \phi_4), \quad \alpha_2 = \sin(\phi_2 + \phi_3 + \phi_4).$$

${}^0\mathbf{R}_e \in \mathbb{R}^{3 \times 3}$  is the *rotation matrix* (another representation for orientation) and  $\mathbf{0}_{1 \times 3} = [0 \ 0 \ 0]$ .

### 2.3.2. Inverse Kinematics

From Figure 4, it is easy to see that orientations in (14) are defined as follows:

$$\zeta_{ef} = \theta - \phi_1 \quad (18)$$

$$\psi_{ef} = \phi_2 + \phi_3 + \phi_4. \quad (19)$$

where  $\theta$  can be written in terms of mobile cartesian velocities—computing  $\theta$  could be tricky due to the  $\tan^{-1}(\bullet)$  function. In order to avoid quadrant problems and inconsistencies (such as having  $\dot{x} = 0$ ), it is recommended to use  $\text{atan2}(\bullet, \bullet)$  function instead—, i.e.,

$$\theta = \tan^{-1}\left(\frac{\dot{y}}{\dot{x}}\right). \quad (20)$$

A simple way for solving the inverse kinematics is defining a desired pose vector  $\mathbf{X}_{ed} = [x_{ef_d} \ y_{ef_d} \ z_{ef_d} \ \psi_{ef_d} \ \zeta_{ef_d}]^\top$  for the end-effector and a desired cartesian position for the mobile base  $[x_d \ y_d]^\top$ . This way,  $\theta$  can be obtained in desired velocities terms and the desired orientation  $\zeta_{ef_d}$  can be used to obtain  $\phi_1$  in (18). Setting (17) equal to the end-effector desired position  $[x_{ef_d} \ y_{ef_d} \ z_{ef_d}]^\top$  and using the desired orientation  $\psi_{ef_d}$  in (19), we obtain a system of non-linear equations that are numerically solved for  $\phi_2, \phi_3$  and  $\phi_4$ .

Finally, note that  $\phi_5$  does not affect the end-effector position nor orientation ( $\phi_5$  is used for rotating the wrist). We can simply establish  $\phi_5$  as a constant.

### 2.3.3. Dynamical Model

The dynamical model is such that  $n = 8$  ( $n_b = 3$  and  $n_m = 5$ ). The detailed definitions of the KUKA youBot model can be found in [18].

### 3. Trajectory Tracking Problem

Let  $\mathbf{q}_d(t)$  and  $\eta_d(t)$  be the desired position and velocity vectors, namely,

$$\mathbf{q}_d(t) = [q_{1d} \ q_{2d} \ \cdots \ q_{nd}]^T = [x_d \ y_d \ \theta_d \ \phi_{1d} \ \cdots \ \phi_{n_m d}]^T \quad (21)$$

$$\eta_d(t) = \mathbf{S}^T(\mathbf{q}_d)\dot{\mathbf{q}}_d = [\eta_{1d} \ \eta_{2d} \ \cdots \ \eta_{(n-1)d}]^T = [v_d \ \dot{\theta}_d \ \dot{\phi}_{1d} \ \cdots \ \dot{\phi}_{n_m d}]^T, \quad (22)$$

to satisfy the non-holonomic constraint, in terms of desired values, we obtain

$$\theta_d = \tan^{-1}\left(\frac{\dot{y}_d}{\dot{x}_d}\right), \quad (23)$$

the mobile velocity references are:

$$v_d = \sqrt{\dot{x}_d^2 + \dot{y}_d^2}, \quad \dot{\theta}_d = \frac{\dot{x}_d \ddot{y}_d - \dot{y}_d \ddot{x}_d}{v_d^2}. \quad (24)$$

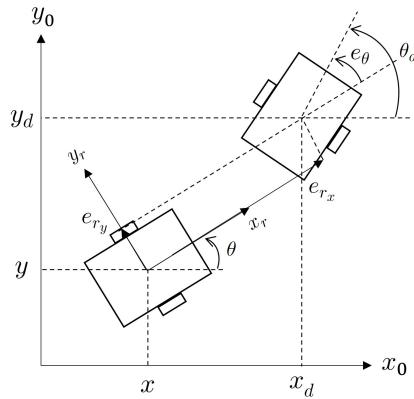
The desired trajectories are designed to be *smooth* functions, i.e., at least twice differentiable and considering that  $v_d \neq 0$ .

Let us define the position and velocity error vectors:

$$\mathbf{e}_q = \mathbf{q}_d - \mathbf{q} = [e_x \ e_y \ e_\theta \ e_{\phi_1} \ \cdots \ e_{\phi_{n_m}}]^T \quad (25)$$

$$\mathbf{e}_\eta = \eta_d - \eta = [e_v \ e_\theta \ e_{\phi_1} \ \cdots \ e_{\phi_{n_m}}]^T. \quad (26)$$

For a differential robot, the trajectory tracking problem can be solved by using a leader-follower scheme, where the leader is proposed to be a virtual robot, and it describes, ideally, the desired trajectory. The objective is that the actual robot converges with the virtual robot [19]. This is represented in Figure 5.



**Figure 5.** Relative errors on differential robot local frame in a leader-follower scheme.

The trajectory tracking error vector, in the actual robot local frame, with respect to the virtual robot, is defined as follows:

$$\mathbf{e}_r = \begin{bmatrix} e_{r_x} \\ e_{r_y} \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta & 0 \\ \sin \theta & \cos \theta & 0 \\ 0 & 0 & 1 \end{bmatrix}^T \begin{bmatrix} x_d - x \\ y_d - y \\ \theta_d - \theta \end{bmatrix}. \quad (27)$$

For an MMR with no interaction with the environment, the generalized trajectory tracking control law is proposed as:

$$\tau_{red} = \mathbf{P}^{-1}(\mathbf{q})\mathbf{H}(\mathbf{q}) \left[ \mathbf{I}^{-1}(\mathbf{q}, \eta) + \dot{\eta}_d + \mathbf{K}_p \mathbf{e}_\eta + \mathbf{U}(\mathbf{q}, \mathbf{q}_d, \eta_d) + \mathbf{G}^*(\mathbf{q}) + \mathbf{F}_f^*(\mathbf{q}, \dot{\mathbf{q}}) \right] \in \mathbb{R}^{n-1}. \quad (28)$$

where:

$$\begin{aligned}\mathbf{U}(\mathbf{q}, \mathbf{q}_d, \eta_d) &= [u_1 \ u_2 \ \cdots \ u_{n-1}]^\top \\ u_1 &= k_1 e_{r_x} \\ u_2 &= k_2 v_d \frac{\sin e_\theta}{e_\theta} e_{r_y} + k_3 e_\theta \\ u_3 &= k_4 e_{\phi_1} \\ &\vdots \\ u_{n-1} &= k_n e_{\phi_{nm}}\end{aligned}$$

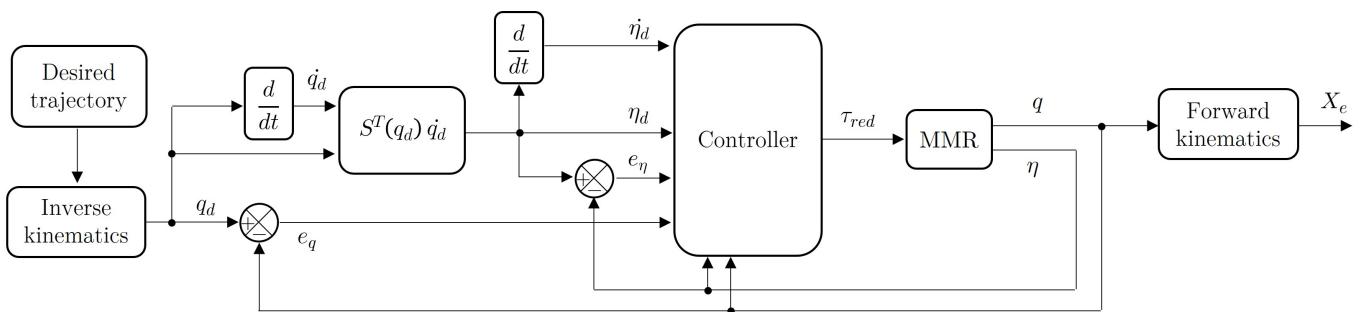
$$\begin{aligned}\mathbf{P}(\mathbf{q}) &= \mathbf{S}^\top(\mathbf{q}) \mathbf{B}(\mathbf{q}), \quad \mathbf{H}(\mathbf{q}) = \bar{\mathbf{D}}(\mathbf{q}), \quad \mathbf{I}^{-1}(\mathbf{q}, \eta) = \bar{\mathbf{D}}^{-1}(\mathbf{q}) \bar{\mathbf{C}}(\mathbf{q}, \dot{\mathbf{q}}) \eta \\ \mathbf{G}^*(\mathbf{q}) &= \bar{\mathbf{D}}^{-1}(\mathbf{q}) \bar{\mathbf{G}}(\mathbf{q}), \quad \mathbf{F}_f^*(\mathbf{q}, \dot{\mathbf{q}}) = \bar{\mathbf{D}}^{-1}(\mathbf{q}) \bar{\mathbf{F}}_f(\mathbf{q}, \dot{\mathbf{q}}).\end{aligned}$$

With gains:

$$k_i > 0, \quad i = 1, \dots, n.$$

$\mathbf{K}_p = \text{diag}\{k_{p_1} \ k_{p_2} \ \dots \ k_{p_{n-1}}\} > 0 \in \mathbb{R}^{(n-1) \times (n-1)}$  is a diagonal positive definite matrix,  $\mathbf{U}(\mathbf{q}, \mathbf{q}_d, \eta_d)$  is the vector in charge of correcting the position error,  $\mathbf{G}^*(\mathbf{q})$  is the gravity compensation term, and  $\mathbf{F}_f^*(\mathbf{q}, \dot{\mathbf{q}})$  compensates friction effects.

The block diagram of the controlled system is shown in Figure 6.



**Figure 6.** Proposed control block diagram for MMR.

### 3.1. Closed-Loop Dynamics

Let

$$\mathbf{e}_\Gamma = \begin{bmatrix} \mathbf{e}_\beta \\ \mathbf{e}_\mu \end{bmatrix}, \quad (29)$$

be the complete error vector. Where the mobile platform error vector is represented by  $\mathbf{e}_\beta$ , and the manipulator error vector is denoted by  $\mathbf{e}_\mu$ , i.e.,

$$\mathbf{e}_\beta = [e_{r_x} \ e_{r_y} \ e_\theta \ \dot{e}_\theta \ e_v]^\top \quad (30)$$

$$\mathbf{e}_\mu = [e_{\phi_1} \ \dot{e}_{\phi_1} \ e_{\phi_2} \ \dot{e}_{\phi_2} \ \cdots \ e_{\phi_{nm}} \ \dot{e}_{\phi_{nm}}]^\top. \quad (31)$$

The closed-loop dynamics are obtained by substituting (28) into the reduced model (13); for the manipulator, we obtain:

$$\ddot{e}_{\phi_i} + k_{p_{i+2}} \dot{e}_{\phi_i} + k_{i+3} e_{\phi_i} = 0, \quad i = 1, \dots, n_m, \quad (32)$$

(32) represents a set of  $n_m$  linear and decoupled systems. Considering  $k_{p_{i+2}}, k_{i+3} > 0$ , it is easy to prove asymptotic stability for states  $[e_{\phi_i} \ \dot{e}_{\phi_i}]^\top$  at the origin, i.e.,  $[e_{\phi_i} \ \dot{e}_{\phi_i}]^\top \rightarrow [0 \ 0]^\top$  as  $t \rightarrow \infty$ , corresponding to the manipulator angular positions and velocities.

However, it is not that clear for the mobile base, whose closed-loop dynamics are represented by the set of equations:

$$\begin{cases} \dot{e}_v + k_{p_1} e_v + k_1 e_{r_x} = 0 \\ \ddot{e}_\theta + k_{p_2} \dot{e}_\theta + k_2 v_d \frac{\sin(e_\theta)}{e_\theta} e_{r_y} + k_3 e_\theta = 0. \end{cases} \quad (33)$$

### 3.2. Error Dynamics

Using Equations (25)–(27) and (2), it is possible to obtain:

$$\begin{cases} \dot{e}_x = \dot{x}_d + (e_v - v_d) \cos(\theta_d - e_\theta) \\ \dot{e}_y = \dot{y}_d + (e_v - v_d) \sin(\theta_d - e_\theta) \\ \dot{e}_{r_x} = v_d \cos(e_\theta) + (\dot{\theta}_d - \dot{e}_\theta) e_{r_y} + e_v - v_d \\ \dot{e}_{r_y} = (\dot{e}_\theta - \dot{\theta}_d) e_{r_x} + v_d \sin(e_\theta) \\ \dot{e}_\theta = e_{\eta_2} \\ \dot{e}_{\phi_i} = e_{\eta_{i+2}}, \quad i = 1, \dots, n_m. \end{cases} \quad (34)$$

The so-called *error dynamics* conforms the set of Equations (32)–(34).

### 3.3. Stability Analysis

Note that there is no need to prove stability for the manipulator due to the linearization achieved in the closed-loop dynamics section, see Section 3.1 (in any case, we provide some proof of this in Appendix A). In the following, the mobile base stability is proven.

Let  $V(\mathbf{e}_\beta)$  be the proposed Lyapunov candidate function:

$$V(\mathbf{e}_\beta) \triangleq \frac{1}{2} k_1 (e_{r_x}^2 + e_{r_y}^2) + \frac{1}{2} (k_3 e_\theta^2 + \dot{e}_\theta^2 + e_v^2). \quad (35)$$

The time derivative of (35), considering the error dynamics, can be written as:

$$\dot{V}(\mathbf{e}_\beta) = k_1 v_d e_{r_x} (\cos(e_\theta) - 1) + k_1 v_d e_{r_y} \sin(e_\theta) - k_2 v_d \frac{\sin(e_\theta)}{e_\theta} \dot{e}_\theta e_{r_y} - k_{p_2} \dot{e}_\theta^2 - k_{p_1} e_v^2. \quad (36)$$

The goal is to prove:

$$\dot{V}(\mathbf{e}_\beta) \leq \|k_1 v_d e_{r_x} (\cos(e_\theta) - 1)\| + \left\|k_1 v_d e_{r_y} \sin(e_\theta)\right\| + \left\|k_2 v_d \frac{\sin(e_\theta)}{e_\theta} \dot{e}_\theta e_{r_y}\right\| - k_{p_2} \|\dot{e}_\theta\|^2 - k_{p_1} \|e_v\|^2 \leq 0. \quad (37)$$

Using that

$$\|e_{r_x}\| \leq \|\mathbf{e}_\beta\|, \quad \|e_{r_y}\| \leq \|\mathbf{e}_\beta\|, \quad \|\dot{e}_\theta\| \leq \|\mathbf{e}_\beta\|, \quad \|e_v\| \leq \|\mathbf{e}_\beta\|,$$

an upper bound is found for every term in (37), i.e.,

$$\begin{aligned} k_1 \|v_d\| \|e_{r_x}\| \|\cos(e_\theta) - 1\| &\leq 2k_1 \|v_d\| \|\mathbf{e}_\beta\| \\ k_1 \|v_d\| \|e_{r_y}\| \|\sin(e_\theta)\| &\leq k_1 \|v_d\| \|\mathbf{e}_\beta\| \\ k_2 \|v_d \frac{\sin(e_\theta)}{e_\theta} \dot{e}_\theta e_{r_y}\| &\leq k_2 \|v_d\| \|\mathbf{e}_\beta\|^2 \\ k_{p_2} \|\dot{e}_\theta\|^2 &\leq k_{p_2} \|\mathbf{e}_\beta\|^2 \\ k_{p_1} \|e_v\|^2 &\leq k_{p_1} \|\mathbf{e}_\beta\|^2 \end{aligned}$$

Then, (37) is re-written as:

$$\dot{V}(\mathbf{e}_\beta) \leq 3k_1\|v_d\|\|\mathbf{e}_\beta\| + (k_2\|v_d\| - k_{p_2} - k_{p_1})\|\mathbf{e}_\beta\|^2 \quad (38)$$

To assure that  $\dot{V}(\mathbf{e}_\beta) \leq 0$ , the following conditions are established:

**Condition 1.** Gains can be chosen such that:

$$k_2\|v_d\| - k_{p_2} - k_{p_1} < 0. \quad (39)$$

**Condition 2.** From the right hand side of (38), and considering that Condition 1 is satisfied, we have to assure that the right term is greater than the left term in magnitude; this yields an interval for the mobile base error vector norm of

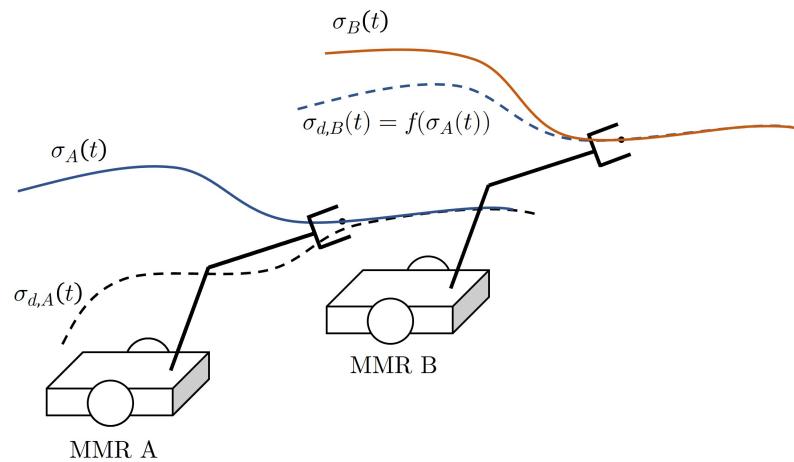
$$\|\mathbf{e}_\beta\| \leq \frac{3k_1\|v_d\|}{k_{p_2} + k_{p_1} - k_2\|v_d\|}. \quad (40)$$

In conclusion, we cannot guarantee asymptotic stability but rather we have an upper bound for the mobile base error norm by a positive quantity; this is termed *UUB stability* [20], p. 85.

Furthermore, (40) denotes a compact set, which can be made arbitrarily large by adjusting the control gains, where the system will be stable. Therefore, *semiglobal stabilization* is achieved [20], pp. 198–199.

#### 4. Synchronization Problem

Two robots A and B in a master–slave configuration, being A the “Master” and B the “Slave”, are considered, see Figure 7.



**Figure 7.** Master–slave scheme.

Let  $\sigma_{d,i}(t) = [\mathbf{q}_{d,i}^\top \ \eta_{d,i}^\top]^\top$  and  $\sigma_i(t) = [\mathbf{q}_i^\top \ \eta_i^\top]^\top$  be desired and actual trajectories for the  $i$ -th robot, respectively, where:

$$\mathbf{q}_{d,i} = \begin{bmatrix} x_{d,i} \\ y_{d,i} \\ \theta_{d,i} \\ \phi_{1d,i} \\ \vdots \\ \phi_{n_{md},i} \end{bmatrix}, \quad \mathbf{q}_i = \begin{bmatrix} x_i \\ y_i \\ \theta_i \\ \phi_{1,i} \\ \vdots \\ \phi_{n_m,i} \end{bmatrix}, \quad \eta_{d,i} = \begin{bmatrix} v_{d,i} \\ \dot{\theta}_{d,i} \\ \dot{\phi}_{1d,i} \\ \vdots \\ \dot{\phi}_{n_{md},i} \end{bmatrix}, \quad \eta_i = \begin{bmatrix} v_i \\ \dot{\theta}_i \\ \dot{\phi}_{1,i} \\ \vdots \\ \dot{\phi}_{n_m,i} \end{bmatrix}, \quad \text{for } i = A, B. \quad (41)$$

In this configuration, the master robot A is controlled independently of the movement of the slave robot B. For the master robot, the trajectory tracking controller posted in the previous section is used for tracking the desired trajectory  $\sigma_{d,A}$ . Meanwhile, the slave reference is formed as a function of the master actual trajectories, i.e.,

$$\sigma_{d,B}(t) = f(\sigma_A(t)), \quad (42)$$

this constitutes a uni-directional information flow (information is passed from master to slave but not viceversa). In order to ensure feedback to the master, coupling errors for the  $i$ -th robot with respect to the  $j$ -th robot are defined:

$${}^j\epsilon_i \in \mathbb{R}^n = \begin{bmatrix} {}^j e_{c_x,i} \\ {}^j e_{c_y,i} \\ {}^j e_{c_\theta,i} \\ {}^j e_{c_{\phi_1},i} \\ {}^j e_{c_{\phi_2},i} \\ \vdots \\ {}^j e_{c_{\phi_{n_m}},i} \end{bmatrix} = \begin{bmatrix} e_{r_x,i} - e_{r_x,j} \\ e_{r_y,i} - e_{r_y,j} \\ e_{\theta,i} - e_{\theta,j} \\ e_{\phi_1,i} - e_{\phi_1,j} \\ e_{\phi_2,i} - e_{\phi_2,j} \\ \vdots \\ e_{\phi_{n_m},i} - e_{\phi_{n_m},j} \end{bmatrix}, \quad i \neq j. \quad (43)$$

In a general framework, with more than two robots in the system, let us define  $Y$  as the set of robots in the whole system;  $\Omega_i \subset Y$  is the subset in which the  $i$ -th robot and its neighbors with whom it shares information are found, and let  $N_i$  be the number of elements in  $\Omega_i$ , i.e.,  $\text{card}(\Omega_i) = N_i$ , with  $\text{card}(\Omega_i) \leq \text{card}(Y)$ . Therefore, the proposed synchronization control strategy for the  $i$ -th robot is defined as:

$$\tau_{\text{red},i} = \mathbf{P}^{-1}(\mathbf{q}_i)\mathbf{H}(\mathbf{q}_i) \left[ \mathbf{I}^{-1}(\mathbf{q}_i, \eta_i) + \dot{\eta}_{d,i} + \mathbf{K}_p \mathbf{e}_{\eta_i} + \mathbf{G}^*(\mathbf{q}_i) + \mathbf{F}_f^*(\mathbf{q}_i, \dot{\mathbf{q}}_i) + \mathbf{U}(\mathbf{q}_i, \mathbf{q}_{d,i}, \eta_{d,i}) + \sum_{j \in \Omega_i} {}^j \mathbf{U}_{s,i} \right], \quad i \neq j. \quad (44)$$

where:

$${}^j \mathbf{U}_{s,i} \in \mathbb{R}^{n-1} = \Lambda_i(\mathbf{e}_{\mathbf{q}_i}) {}^j \mathbf{C}_i {}^j \epsilon_i \quad (45)$$

$$\Lambda_i(\mathbf{e}_{\mathbf{q}_i}) \in \mathbb{R}^{(n-1) \times n} = \begin{bmatrix} 1 & 0 & 0 & \mathbf{0}_{1 \times n_m} \\ 0 & \frac{\sin(e_{\theta,i})}{e_{\theta,i}} & 1 & \mathbf{0}_{1 \times n_m} \\ \mathbf{0}_{n_m \times 1} & \mathbf{0}_{n_m \times 1} & \mathbf{0}_{n_m \times 1} & \mathbf{1}_{n_m \times n_m} \end{bmatrix}$$

$${}^j \mathbf{C}_i \in \mathbb{R}^{n \times n} = \text{diag}\{ {}^j C_{x,i} \ {}^j C_{y,i} \ {}^j C_{\theta,i} \ {}^j C_{\phi_1,i} \ \dots \ {}^j C_{\phi_{n_m},i} \}$$

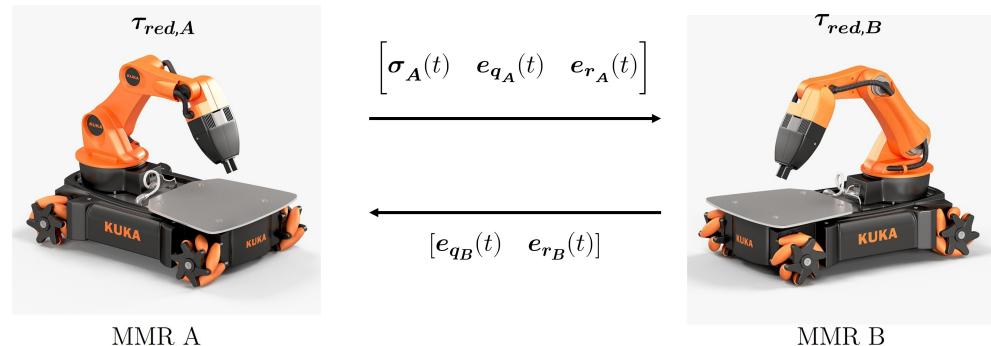
$$\mathbf{e}_{\mathbf{q}_i} = \mathbf{q}_{d,i} - \mathbf{q}_i$$

$$\mathbf{e}_{\eta_i} = \eta_{d,i} - \eta_i.$$

${}^j \mathbf{U}_{s,i}$  is the term that contains the static couplings for the  $i$ -th robot with respect to the  $j$ -th robot; this term is in charge to achieve synchronization.  ${}^j \mathbf{C}_i$  is the positive definite diagonal matrix of synchronization control gains for the  $i$ -th robot;  $\mathbf{e}_{\mathbf{q}_i}$  and  $\mathbf{e}_{\eta_i}$  are the position and velocity error vectors for the  $i$ -th robot, respectively. The remaining terms are defined in (28) with respect to the  $i$ -th robot and are similar for the  $j$ -th one.

The provided synchronization strategy allows one to design networks of robots working together in different configurations. For the sake of clarity, in this paper only the cooperative manipulation with two robots is kept; the synchronization scheme is shown in Figure 8, where  $\mathbf{e}_{r_i}$ ,  $i = A, B$  are the relative errors for the mobile base of the  $i$ -th robot, see (27). In the following, the MMR position errors are used to obtain the coupling errors, and both errors allow us to compute the control (44) for the  $i$ -th robot.

In the next section, firstly, the control performance for one MMR is experimentally tested for the trajectory tracking problem; subsequently, the master–slave scheme, using two MMRs for the synchronization problem, is simulated under the paradigm of cooperative manipulation, considering a virtual object.

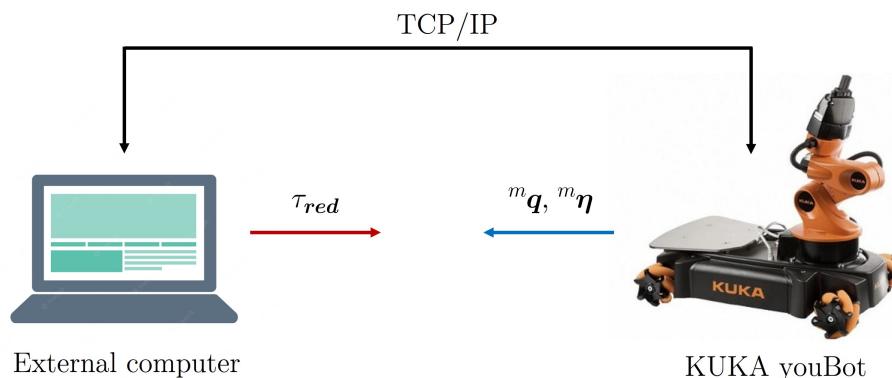


**Figure 8.** Bi-directional flow information between two KUKA youBot robots.

## 5. Results

### 5.1. Trajectory Tracking Experimental Results

Communication with KUKA youBot is achieved using a TCP/IP protocol, through *sockets*. This allowed us to have active communication between a server (external computer) and a client (KUKA youBot), see Figure 9. KUKA youBot uses Ubuntu 12.04 LTS as OS, and the programming language used in the external computer was C++. The sampling time was found to be of 2.8 ms, which is small enough to consider feasible the use of the equations of the continuous system in the discrete domain.

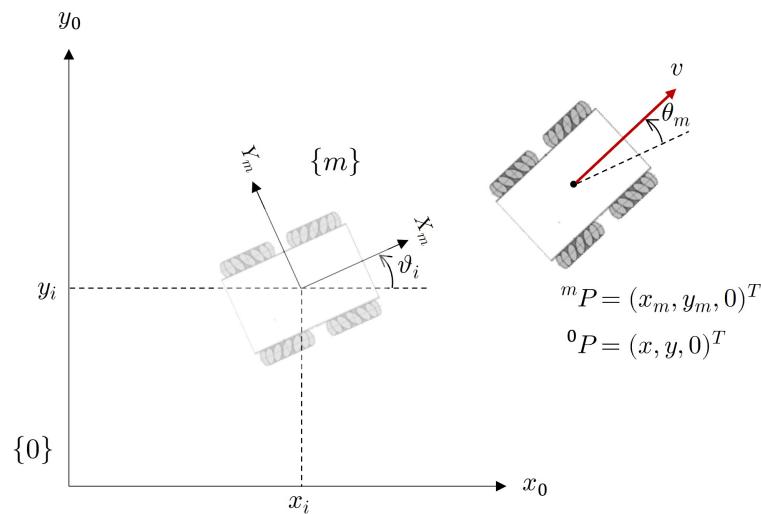


**Figure 9.** Communication with KUKA youBot.

It is actually possible to control the KUKA youBot dynamically, i.e., the robot receives torque commands from the external computer and sends information about its positions and velocities, measured on a frame established by the robot itself, that we denoted as frame  $\{m\}$ , as feedback. To obtain the states needed for computing the control  $\tau_{red}$ , a coordinate transformation was made.

Considering Figure 10, we defined  $(x_i, y_i, \theta_i)$  as the initial pose for the KUKA youBot mobile base on the inertial frame  $\{0\}$ . The initial pose for the KUKA youBot mobile base is constant (i.e.,  $(x_i, y_i, \theta_i)$  are not time-varying), where the robot defines its own inertial frame named  $\{m\}$ .

The robot sends the states  $(x_m, y_m, \theta_m)$  to the computer, which correspond to the position and orientation for the mobile base on the horizontal plane measured with respect to  $\{m\}$ , and to longitudinal and angular velocity  $(v, \dot{\theta}_m)$  and  $(\phi_i, \dot{\phi}_i)$ , respectively, with  $i = 1, \dots, 5$  for the manipulator angular positions and velocities.



**Figure 10.** Inertial frames and coordinates relationship.

To obtain the states measured on inertial frame  $\{0\}$ , the next transformation was considered:

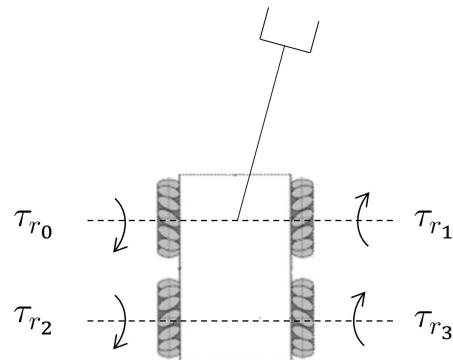
$$\begin{cases} x = x_i + x_m \cos \vartheta_i - y_m \sin \vartheta_i \\ y = y_i + y_m \cos \vartheta_i + x_m \sin \vartheta_i \\ \dot{x} = \dot{x}_m \cos \vartheta_i - \dot{y}_m \sin \vartheta_i \\ \dot{y} = \dot{y}_m \cos \vartheta_i + \dot{x}_m \sin \vartheta_i \\ \theta = \vartheta_i + \theta_m \\ \dot{\theta} = \dot{\vartheta}_m. \end{cases} \quad (46)$$

where:

$$\dot{x}_m = v \cos(\theta_m), \quad \dot{y}_m = v \sin(\theta_m).$$

According with the first assumption made on Section 2.3 for the mobile base, namely, *the two wheels of each side rotate at the same speed*; the next torques relation is considered, see Figure 11:

$$\tau_{r0} = \tau_{r2} = -\frac{\tau_i}{2}, \quad \tau_{r1} = \tau_{r3} = \frac{\tau_d}{2}. \quad (47)$$



**Figure 11.** Torque positive directions on robot wheels.

Control actions were bounded (in code) as follows:

$$\begin{aligned} -10 \text{ [N} \cdot \text{m}] &\leq \tau_d, \tau_i, \tau_5 \leq 10 \text{ [N} \cdot \text{m}] \\ -15 \text{ [N} \cdot \text{m}] &\leq \tau_1, \tau_2, \tau_3, \tau_4 \leq 15 \text{ [N} \cdot \text{m}]. \end{aligned}$$

The desired trajectory is defined as a circle with center at  $(x_c, y_c) = (-1, 0)$ , radius  $r_d = 1$  for the mobile base, and radius  $r_{ef} = 1.3$  for the end-effector, i.e.,

$$\begin{cases} \delta(t) = \frac{2\pi}{t_f} t - \sin\left(\frac{2\pi}{t_f} t\right) \\ x_d(t) = x_c + r_d \cos(\delta(t)), \\ y_d(t) = y_c + r_d \sin(\delta(t)) \end{cases} \quad \begin{cases} \delta_{ef}(t) = \delta(t) + \sin^{-1}\left(\frac{L_x}{r_{ef}}\right) \\ x_{ef_d}(t) = x_c + r_{ef} \cos(\delta_{ef}(t)) \\ y_{ef_d}(t) = y_c + r_{ef} \sin(\delta_{ef}(t)) \\ z_{ef_d}(t) = 0.46. \end{cases} \quad (48)$$

This planning was thought for a time  $t_f = 24$  s.  
 $\theta_d$  is computed as follows:

$$\theta_d(t) = \begin{cases} \pi/2 + 2\pi n_c, & \text{if } \dot{x}_d, \dot{y}_d = 0 \\ \text{atan2}(\dot{y}_d, \dot{x}_d) + 2\pi n_c, & \text{if } (\dot{y}_d > 0) \text{ and } (\dot{x}_d < 0) \\ \text{atan2}(\dot{y}_d, \dot{x}_d) + 2\pi + 2\pi n_c, & \text{if } (\dot{y}_d < 0) \text{ or } (\dot{x}_d, \dot{y}_d > 0). \end{cases} \quad (49)$$

where  $n_c$  is the number of cycles given by the mobile base. For this experiment, just one cycle is considered, and this counter takes the value of  $n_c = 1$  at the end of the trajectory, i.e., at  $t = t_f$ .

Notice that the proposed trajectory starts and finishes with  $v_d = 0$ . To avoid singularities for  $\dot{\theta}_d(t)$  in (24), it is possible to set  $\dot{\theta}_d(t) = 0$  when  $v_d(t) = 0$  artificially, in code; however, we cannot expect the stability condition (40) to be satisfied when this happens, see Figure 12.

The end-effector desired orientations are defined as:

$$\psi_{ef_d}(t) = \frac{\pi}{2}, \quad \zeta_{ef_d}(t) = \theta_d - \frac{\pi}{2}. \quad (50)$$

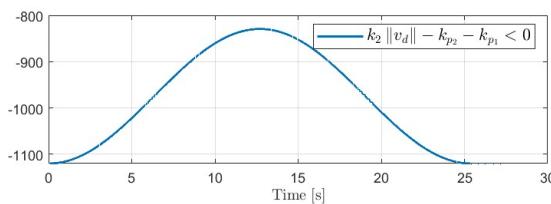
All the initial conditions were set to 0, except for  $\theta(0) = \pi/2$ .

Finally, with the parameters shown in Table A1 (these parameters are not exactly known; uncertainties exist, and some of them where calculated or supposed), and control gains are shown in Table 1; performance charts were obtained, see Figure 13.

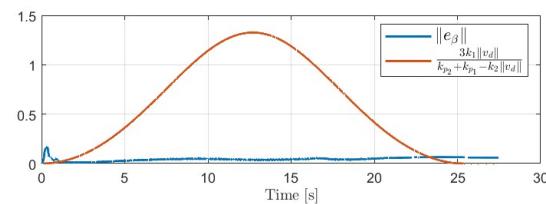
The video of the experiment is shown in the following link: <https://drive.google.com/file/d/1We9XskIA2-GLTUxg973DUMGcWTJm1zAz/view>, accessed on 7 Septembre 2023.

**Table 1.** Trajectory tracking control gains.

$k_1$	$k_2$	$k_3$	$k_4$	$k_5$	$k_6$	$k_7$	$k_8$	$k_{p_1}$	$k_{p_2}$	$k_{p_3}$	$k_{p_4}$	$k_{p_5}$	$k_{p_6}$	$k_{p_7}$
700	555	300	1000	1000	2000	2500	100	400	720	350	350	350	350	350

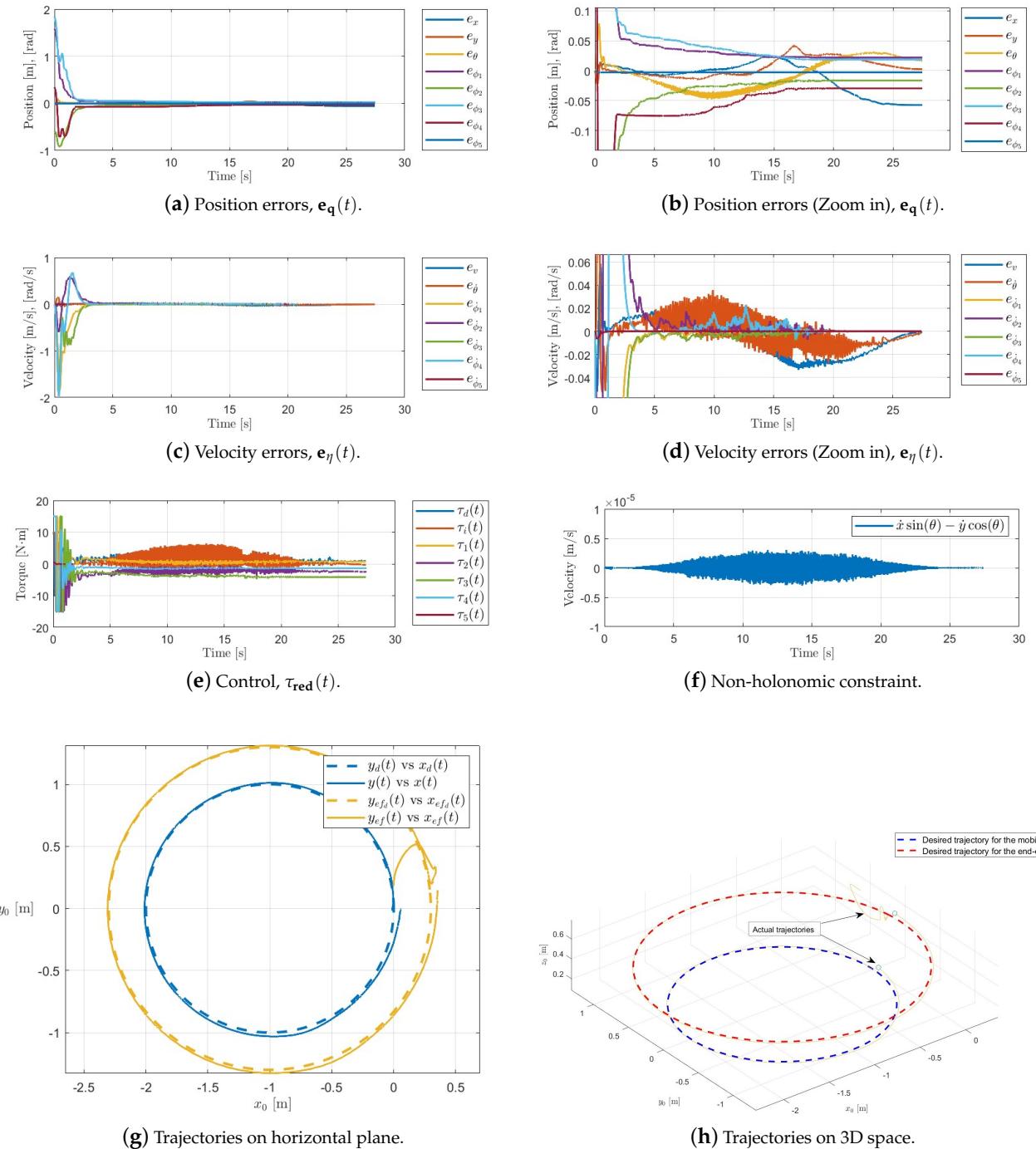


(a) Condition 1



(b) Condition 2

**Figure 12.** Stability conditions: numerical verification.



**Figure 13.** KUKA youBot performance.

### 5.2. Synchronization: Numerical Simulations

In this simulation, two KUKA youBot robots (A and B) with  $n = 8$  DoF, being A the master and B the slave, are considered. Coupling errors shown in (43) and control (44) with  $i, j = A, B$  were used.

The task is about manipulating a 20 cm side length virtual box along a circular trajectory, i.e., robots have to guarantee the grip of an object maintaining the distance among the end-effectors while the trajectory is executed. Emphasizing on the word *virtual*, in this work, the interaction with the environment is neglected, and the term  $\mathbf{J}_e^T(\mathbf{q})\mathbf{h}_r$ , in models (6) and (13), is considered zero.

The trajectory planning is defined by four stages:

- **The first stage:** The robot mobile base describes a straight line until the box position is reached, and the manipulator is positioned in front of the object.
- **The second stage:** In this stage, the end-effector makes contact with the object in a smooth manner. It is considered a punctual grip (without friction); thus, a perpendicular grip with the object is required.
- **The third stage:** This stage is about maintaining the position for a few seconds in order to wait for the effects of tapping that may exist between the object and the end-effector to pass, and for the grip to reach a point of stability.
- **The fourth stage:** The robots generate the desired circle.

The object position is considered at  $(4.5, 2, 0.46)$  on frame  $\{0\}$ , and the initial position of the mobile base is at  $(4.5, 0)$ . The trajectory planning is defined as follows:

**The first stage,  $0 \leq t \leq 8$  s:**

For robot A, the desired trajectories are defined as:

$$\begin{cases} p_1(t) = -\frac{5t^7}{524288} + \frac{35t^6}{131072} - \frac{21t^5}{8192} + \frac{35t^4}{4096} \\ x_{d,A}(t) = 4 \\ y_{d,A}(t) = (2 - L_x)p_1(t) \end{cases}, \quad \begin{cases} x_{ef_d,A}(t=8) = 4.35 \\ y_{ef_d,A}(t=8) = y_{d,A}(t=8) + L_x \\ z_{ef_d,A}(t=8) = 0.46 \\ \psi_{ef_d,A}(t=8) = \frac{\pi}{2} \\ \zeta_{ef_d,A}(t=8) = 0. \end{cases} \quad (51)$$

To obtain reasonable control actions in the manipulator, the function  $p_1(t)$  is used to interpolate the trajectory from the initial point to the last, softly, for every link, i.e.,

$$\begin{cases} \phi_{1d,A}(t) = \frac{\pi}{2}p_1(t) \\ \phi_{2d,A}(t) = -0.222635p_1(t) \\ \phi_{3d,A}(t) = 1.63105p_1(t) \\ \phi_{4d,A}(t) = 0.162383p_1(t). \end{cases}$$

For robot B, the desired trajectories are defined as:

$$\mathbf{q}_{d,B} = \mathbf{W}_q^{s_{1,2,3}} \mathbf{q}_A + \mathbf{Q}_q^{s_{1,2,3}}, \quad \boldsymbol{\eta}_{d,B} = \mathbf{W}_\eta^{s_{1,2,3}} \boldsymbol{\eta}_A. \quad (52)$$

where:

$$\mathbf{W}_q^{s_{1,2,3}} = \text{diag}\{1 \ 1 \ 1 \ -1 \ 1 \ 1 \ 1 \ 1\}, \quad \mathbf{W}_\eta^{s_{1,2,3}} = \text{diag}\{1 \ 1 \ -1 \ 1 \ 1 \ 1 \ 1\}, \quad \mathbf{Q}_q^{s_{1,2,3}} = [1 \ \mathbf{0}_{1 \times 7}]^\top.$$

**The second stage,  $8 < t \leq 12$  s:**

For robot A, the desired trajectories are defined as follows:

$$\begin{cases} x_{d,A}(t) = 4 \\ y_{d,A}(t) = 2 - L_x \end{cases}, \quad \begin{cases} p_2(t) = 35(\frac{t}{4} - 2)^4 - 84(\frac{t}{4} - 2)^5 + 70(\frac{t}{4} - 2)^6 - 20(\frac{t}{4} - 2)^7 \\ x_{ef_d,A}(t) = (4.4 + \rho_d - 4.35)p_2(t) + 4.35 \\ y_{ef_d,A}(t) = 2 \\ z_{ef_d,A}(t) = 0.46 \\ \psi_{ef_d,A}(t) = \frac{\pi}{2} \\ \zeta_{ef_d,A}(t) = 0. \end{cases} \quad (53)$$

where  $\rho_d$  is a desired penetration on the virtual object to guarantee the grip.

For robot B, the desired trajectories are defined as in Equation (52).

**The third stage,  $12 < t \leq 20$  s:**

For robot A, the desired trajectories are defined as follows:

$$\begin{cases} x_{d,A}(t) = 4 \\ y_{d,A}(t) = 2 - L_x \end{cases}, \quad \begin{cases} x_{ef_d,A}(t) = 4.4 + \rho_d \\ y_{ef_d,A}(t) = 2 \\ z_{ef_d,A}(t) = 0.46 \\ \psi_{ef_d,A}(t) = \frac{\pi}{2} \\ \zeta_{ef_d,A}(t) = 0. \end{cases} \quad (54)$$

For robot B, the desired trajectories are defined as in Equation (52).

**The fourth stage**,  $20 < t \leq t_f = 60$  s:

For this stage, a circle with center at  $(x_c, y_c) = (3, 2 - L_x)$  and radius  $r_{d,A} = 1$  m on the horizontal plane for the robot A mobile base is considered, and the end-effector has a radius  $r_{ef,A} = \sqrt{L_x^2 + (r_{d,A} + 0.4 + \rho_d)^2}$ . Therefore, the desired trajectories for robot A are defined as:

$$\begin{cases} \delta_A(t) = \frac{2\pi}{(t_f-20)}(t-20) - \sin\left(\frac{2\pi}{(t_f-20)}(t-20)\right), \\ x_{d,A}(t) = x_c + r_{d,A} \cos(\delta_A(t)) \\ y_{d,A}(t) = y_c + r_{d,A} \sin(\delta_A(t)) \end{cases}, \quad \begin{cases} \delta_{ef_d,A}(t) = \delta_A(t) + \sin^{-1}\left(\frac{L_x}{r_{ef,A}}\right) \\ x_{ef_d,A}(t) = x_c + r_{ef,A} \cos(\delta_{ef_d,A}(t)) \\ y_{ef_d,A}(t) = y_c + r_{ef,A} \sin(\delta_{ef_d,A}(t)) \\ z_{ef_d,A}(t) = 0.46 \\ \psi_{ef_d,A}(t) = \frac{\pi}{2} \\ \zeta_{ef_d,A}(t) = \theta_{d,A}(t) - \frac{\pi}{2}. \end{cases} \quad (55)$$

For the robot B mobile base, a circle with the same center  $(x_c, y_c) = (3, 2 - L_x)$  but with radius  $r_{d,B} = r_{d,A} + 1$  is considered. The desired trajectories are defined as:

$$\mathbf{q}_{d,B} = \mathbf{Q}_q^{s_4} + \mathbf{W}_q^{s_4}(\mathbf{q}_A - \mathbf{Q}_q^{s_4}), \quad \boldsymbol{\eta}_{d,B} = \mathbf{W}_\eta^{s_4} \boldsymbol{\eta}_A. \quad (56)$$

where:

$$\mathbf{W}_q^{s_4} = \text{diag}\left\{\frac{r_{d,B}}{r_{d,A}}, \frac{r_{d,B}}{r_{d,A}}, 1, -1, 1, 1, 1, 1\right\}, \quad \mathbf{W}_\eta^{s_4} = \text{diag}\left\{\frac{r_{d,B}}{r_{d,A}}, 1, -1, 1, 1, 1, 1\right\}, \quad \mathbf{Q}_q^{s_4} = [x_c \ y_c \ \mathbf{0}_{1 \times 6}]^\top$$

In general,  $\phi_{5d,A}(t) = 0$  for all  $t$  and  $\theta_{d,A}(t)$  is computed using (49). The functions  $p_1(t)$  and  $p_2(t)$  are soft polynomials, generated with the method based on Bezier curves presented in [21], such that their first time-derivative at the beginning and at the end of the trajectory is zero.

Initial conditions are considered as 0 for positions ( $q_i(0)$ ) and velocities ( $\dot{q}_i(0)$ ) for both robots,  $i = A, B$ , except for the mobile base position: for robot A,  $(x_A(0), y_A(0), \theta_A(0)) = (4, 0, \pi/2)$  is considered; and for robot B,  $(x_B(0), y_B(0), \theta_B(0)) = (5, 0, \pi/2)$ .

The control gains for both robots A and B are shown in Table 2, and the coupling gains are shown in Table 3. A desired penetration of  $\rho_d = 0.002$  m on the object is considered.

Performance charts are shown in Figure 14, and trajectories are presented in Figure 15.

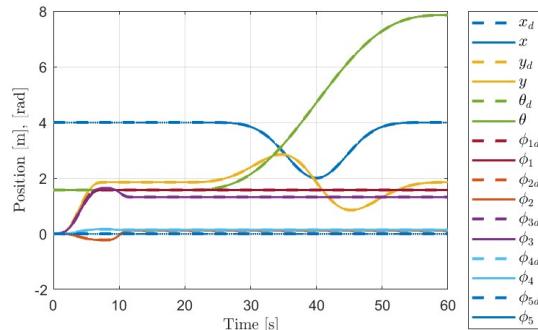
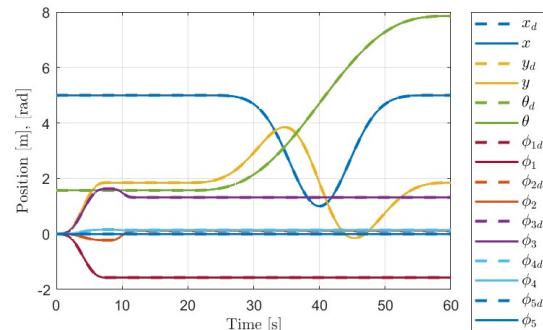
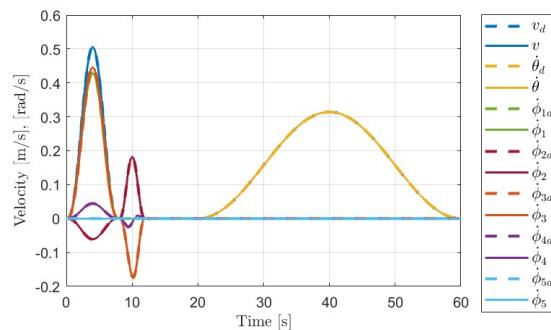
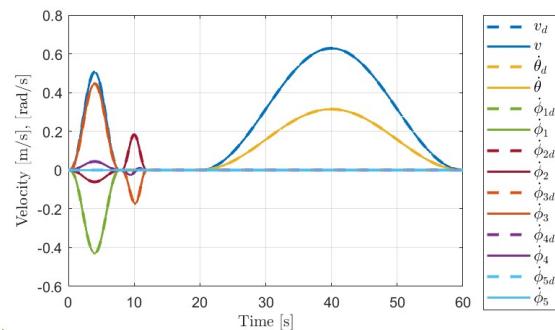
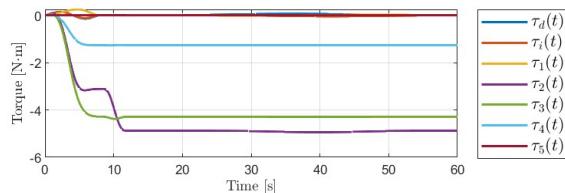
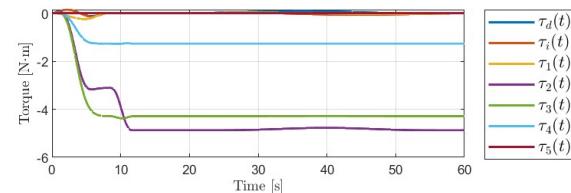
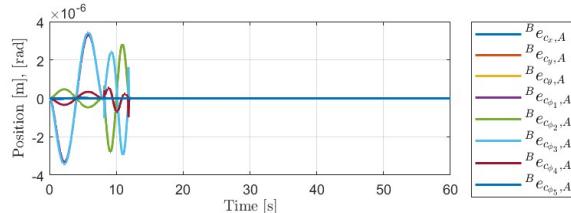
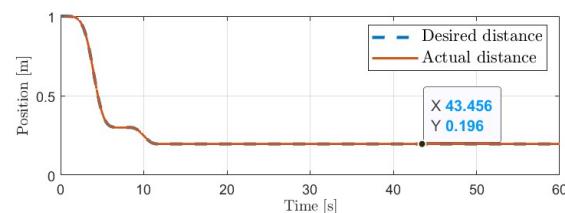
The simulation was animated on CoppeliaSim, and the video is shown in the next link: <https://drive.google.com/file/d/1Z4RWykuw5bglV7KKbzTZKf0beEjUXghi/view?usp=sharing>, accessed on 7 Septembre 2023.

**Table 2.** Control gains for robots A and B.

$k_1$	$k_2$	$k_3$	$k_4$	$k_5$	$k_6$	$k_7$	$k_8$	$k_{p_1}$	$k_{p_2}$	$k_{p_3}$	$k_{p_4}$	$k_{p_5}$	$k_{p_6}$	$k_{p_7}$
50	50	200	400	400	400	400	400	50	50	50	50	50	50	50

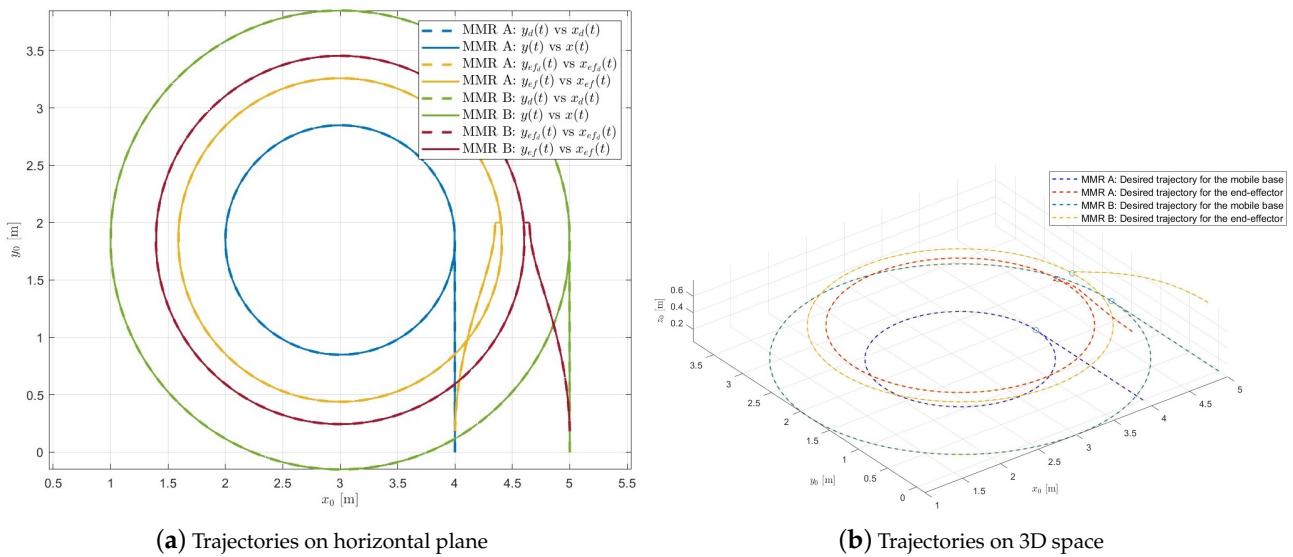
**Table 3.** Coupling gains for robots A and B.

$C_{x,A}$	$C_{y,A}$	$C_{\theta,A}$	$C_{\phi_1,A}$	$C_{\phi_2,A}$	$C_{\phi_3,A}$	$C_{\phi_4,A}$	$C_{\phi_5,A}$	$C_{x,B}$	$C_{y,B}$	$C_{\theta,B}$	$C_{\phi_1,B}$	$C_{\phi_2,B}$	$C_{\phi_3,B}$	$C_{\phi_4,B}$	$C_{\phi_5,B}$
200	200	200	200	200	200	200	200	800	800	800	800	800	800	800	800

(a) MMR A: Position states,  $\mathbf{q}_A(t)$ (b) MMR B: Position states,  $\mathbf{q}_B(t)$ (c) MMR A: Velocity states,  $\eta_A(t)$ (d) MMR B: Velocity states,  $\eta_B(t)$ (e) MMR A: Control,  $\tau_{red,A}(t)$ .(f) MMR B: Control,  $\tau_{red,B}(t)$ .(g) Coupling errors,  $B\epsilon_A = -A\epsilon_B$ .

(h) Distance between end-effectors.

**Figure 14.** KUKA youBot synchronization performance.



**Figure 15.** KUKA youBot trajectories.

## 6. Discussion

A non-linear control law for a  $n$  DoF MMR with semiglobal and UUB stability properties for solving the trajectory tracking problem, considering movement (kinematic) constraints, was designed and tested on the experimental platform KUKA youBot. Reasonable control actions, the accomplishment of the non-holonomic constraint, and small position and velocity errors were achieved with a maximum position error of 6 cm, see Figure 13, even in the presence of parametric uncertainties and unmodeled dynamics. The stability conditions were validated when  $v_d \neq 0$ , see Figure 12; notice that condition 2 is violated when  $v_d = 0$ , as expected.

The trajectory tracking control designed was extended to the synchronization notion, which is valid for both uni-directional or bi-directional information flow between robots, and for any number of robots in a given network through static couplings considering a master–slave configuration. Numerical simulation considering 2 KUKA youBot models and a manipulation task using a 20 cm virtual box was developed in the CoppeliaSim environment. From Figure 14, reasonable control actions and really small errors for both robots can be observed. The grip of the box is guaranteed maintaining a distance between the end-effectors of 19.6 cm, as expected, since the desired penetration ( $\rho_d$ ) on the object of 2mm was considered of for each robot.

The generalized MMR model has been formulated considering the simplest friction model viscous, i.e.,  $f_{f_i} = b_{q_i}\dot{q}_i$  is the  $i$ -th element of  $\mathbf{F}_f(\mathbf{q}, \dot{\mathbf{q}})$  with  $b_{q_i}$  viscous friction coefficients. Friction forces could be up to the 25% of the torque required to move a manipulator [22], as future work is considered to use a more fit friction model as the LuGre model.

Regarding the synchronization approach, experimental validation is expected for future works. Moreover, adding velocity-level coupling terms and changing to dynamical couplings instead of static couplings is also envisioned. In this work, we have supposed a virtual box as a manipulated object to validate the synchronization control on simulation. In the manipulation sense, when robots are interacting with their environment, the term  $\mathbf{J}_e^\top(\mathbf{q})\mathbf{h}_r$  in (6) and (13) becomes relevant for the dynamics of the system; to this end, adding a force controller in the MMR's end-effector print on the object for manipulation purposes will be also desirable, leading to the design of a hybrid position/force controller.

Finally, given the parametric uncertainties and non-modeled dynamics on the system, adding an adaptation law or use a parameter estimator would address the robustness of the controller.

**Author Contributions:** Conceptualization, J.G.P.-F., A.B.M.-D. and A.R.-Á.; methodology, J.G.P.-F., A.B.M.-D. and A.R.-Á.; software, J.G.P.-F.; validation, J.G.P.-F.; formal analysis, J.G.P.-F., A.B.M.-D. and A.R.-Á.; investigation, J.G.P.-F., A.B.M.-D. and A.R.-Á.; writing—original draft preparation, J.G.P.-F.; writing—review and editing, J.G.P.-F., A.B.M.-D. and A.R.-Á.; and supervision, A.B.M.-D. and A.R.-Á. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Consejo Nacional de Ciencia y Tecnología, grant number 1079389; and Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional, grant number IE-ES-2022-123. The authors are also grateful for the support through the project SEP CONACYT CB2017-2018-A1-S-26123: “Análisis, control y sincronización de sistemas complejos con interconexiones dinámicas y acoplamientos flexibles”.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Not applicable.

**Acknowledgments:** The authors are grateful for the support through the project SEP CONACYT CB2017-2018-A1-S-26123.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

MDPI	Multidisciplinary Digital Publishing Institute
MMR	Mobile manipulator robot
UUB	Uniformly and ultimately boundedness
DoF	Degrees of freedom
CoM	Center of mass
OS	Operating system

## Appendix A. Manipulator Stability

Let us re-write the set of Equations (32) for a  $n_m$  DoF manipulator as follows:

$$\ddot{e}_{\phi_i} = -k_{p_{i+2}} \dot{e}_{\phi_i} - k_{i+3} e_{\phi_i}, \quad i = 1, 2, \dots, n_m. \quad (\text{A1})$$

With  $k_{p_{i+2}}, k_{i+3} > 0$  and  $[e_{\phi_i} \quad \dot{e}_{\phi_i}]^T$  constituting the state-vector, where  $e_{\phi_i}$  is the angular position error for the  $i$ -th link and  $\dot{e}_{\phi_i}$  the angular velocity error.

Then, let

$$V(\mathbf{e}_\mu) \triangleq \frac{1}{2} \dot{e}_{\phi_i}^2 + \frac{1}{2} k_{i+3} e_{\phi_i}^2, \quad (\text{A2})$$

the proposed Lyapunov candidate function. Taking the time-derivative of Equation (A2) along the trajectories of the system (A1), we obtain:

$$\dot{V}(\mathbf{e}_\mu) = -k_{p_{i+2}} \dot{e}_{\phi_i}^2 \leq 0. \quad (\text{A3})$$

Since  $k_{p_{i+2}} > 0$ , notice that  $\dot{V}(\mathbf{e}_\mu) = 0$  only when  $\dot{e}_{\phi_i} = 0$ , but the position error vector  $e_{\phi_i}$  could “get stuck” in any other value. In order to analyze this situation, consider the system (A1) when  $\dot{e}_{\phi_i} = 0$ —this implies having  $\ddot{e}_{\phi_i} = 0$ ; then, we obtain:

$$k_{i+3} e_{\phi_i} = 0, \quad i = 1, 2, \dots, n_m. \quad (\text{A4})$$

Since  $k_{i+3} > 0$ , the only solution is  $e_{\phi_i} = 0$ . Thus, the only invariant set that satisfies  $\dot{V} = 0$  is the origin  $[e_{\phi_i} \quad \dot{e}_{\phi_i}]^T = [0 \quad 0]^T$ . Therefore, according to LaSalle, it can be concluded that the system is *globally asymptotically stable*.

## Appendix B. Technical Data for KUKA youBot

Some technical data and parameters for KUKA youBot can be found at [23]; in this paper, limitations on the arm angular positions and velocities have been considered, especially the limitation on the mobile base longitudinal velocity,  $v \leq 0.8$  m/s, which was fundamental for the trajectory planning.

The parameters that were used for experimental validation are shown in Table A1:

**Table A1.** KUKA youBot physical parameters.

Parameter	Value	Parameter	Value	Parameter	Value
$m_m$ [kg]	19.803	$L_{c_b}$ [m]	0.036	$I_{2zz}$ [kg · m <sup>2</sup> ]	0.0031631
$m_b$ [kg]	0.961	$L_{c_1}$ [m]	0.058	$I_{3xx}$ [kg · m <sup>2</sup> ]	0.00041967
$m_1$ [kg]	1.390	$L_{c_2}$ [m]	0.11397	$I_{3yy}$ [kg · m <sup>2</sup> ]	0.00172767
$m_2$ [kg]	1.318	$L_{c_3}$ [m]	0.104	$I_{3zz}$ [kg · m <sup>2</sup> ]	0.0018468
$m_3$ [kg]	0.821	$L_{c_4}$ [m]	0.053	$I_{4xx}$ [kg · m <sup>2</sup> ]	0.0006610
$m_4$ [kg]	0.769	$L_{c_5}$ [m]	0.016	$I_{4yy}$ [kg · m <sup>2</sup> ]	0.0006764
$m_5$ [kg]	0.906	$r_b$ [m]	0.033	$I_{4zz}$ [kg · m <sup>2</sup> ]	0.0010573
$h_b$ [m]	0.030	$r_c$ [m]	0.0255	$I_{5xx}$ [kg · m <sup>2</sup> ]	0.0005563
$h_c$ [m]	0.110	$L_x$ [m]	0.151	$I_{5yy}$ [kg · m <sup>2</sup> ]	0.0003926
$R$ [m]	0.050	$I_{mxx}$ [kg · m <sup>2</sup> ]	0.2657	$I_{5zz}$ [kg · m <sup>2</sup> ]	0.0002756
$p_x$ [m]	0	$I_{myy}$ [kg · m <sup>2</sup> ]	0.5875	$b_x$ [N s/m]	0.8
$p_y$ [m]	0	$I_{mzz}$ [kg · m <sup>2</sup> ]	0.5875	$b_y$ [N s/m]	0.8
$p_z$ [m]	-0.001	$I_{bxx}$ [kg · m <sup>2</sup> ]	0.00041515	$b_\theta$ [N s/m]	0.7
$L$ [m]	0.150	$I_{byy}$ [kg · m <sup>2</sup> ]	0.003553778	$b_{\phi_1}$ [N s/m]	0.5
$L_b$ [m]	0.072	$I_{bzz}$ [kg · m <sup>2</sup> ]	0.00041515	$b_{\phi_2}$ [N s/m]	0.5
$L_1$ [m]	0.075	$I_{1xx}$ [kg · m <sup>2</sup> ]	0.0058821	$b_{\phi_3}$ [N s/m]	0.5
$L_2$ [m]	0.155	$I_{1yy}$ [kg · m <sup>2</sup> ]	0.0029525	$b_{\phi_4}$ [N s/m]	0.5
$L_3$ [m]	0.135	$I_{1zz}$ [kg · m <sup>2</sup> ]	0.0060091	$b_{\phi_5}$ [N s/m]	0.5
$L_4$ [m]	0.081	$I_{2xx}$ [kg · m <sup>2</sup> ]	0.0005843		
$L_5$ [m]	0.137	$I_{2yy}$ [kg · m <sup>2</sup> ]	0.0031145		

## References

- Wang, C.; Sun, D. A synchronization control strategy for multiple robot systems using shape regulation technology. In Proceedings of the 2008 7th World Congress on Intelligent Control and Automation, Chongqing, China, 25–27 June 2008; pp. 467–472. [[CrossRef](#)]
- Markus, E.D. Differential Flatness Based Synchronization Control of Multiple Heterogeneous Robots. In Proceedings of the IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, 21–23 October 2018; pp. 3659–3664. [[CrossRef](#)]
- Fathallah, M.; Abdelhedi, F.; Derbel, N. Synchronization of multi-robot manipulators based on high order sliding mode control. In Proceedings of the 2017 International Conference on Smart, Monitored and Controlled Cities (SM2C), Sfax, Tunisia, 17–19 February 2017; pp. 138–142. [[CrossRef](#)]
- Cicek, E.; Dasdemir, J. Output feedback synchronization of multiple robot systems under parametric uncertainties. *Trans. Inst. Meas. Control* **2017**, *39*, 277–287. [[CrossRef](#)]
- Cicek, E.; Dasdemir, J.; Zergeroglu, E. Coordinated synchronization of multiple robot manipulators with dynamical uncertainty. *Trans. Inst. Meas. Control* **2015**, *37*, 672–683. [[CrossRef](#)]
- Portillo-Vélez, R.d.J.; Cruz-Villar, C.A.; Rodríguez-Ángeles, A. On-line master/slave robot system synchronization with obstacle avoidance. *Stud. Inform. Control* **2012**, *21*, 17–26. [[CrossRef](#)]
- Zhao, X.; Zhang, Y.; Ding, W.; Tao, B.; Ding, H. A Dual-Arm Robot Cooperation Framework Based on a Nonlinear Model Predictive Cooperative Control. *IEEE/ASME Trans. Mechatron.* **2023**, *1*–13. . [[CrossRef](#)]
- Rosales-Hernández, F.; Velasco-Villa, M.; Castro-Linares, R.; del Muro-Cuéllar, B.; Hernández-Pérez, M.Á. Synchronization strategy for differentially driven mobile robots: Discrete-time approach. *Int. J. Robot. Autom.* **2015**, *30*, 50–59. [[CrossRef](#)]
- Sun, D.; Wang, C.; Shang, W.; Feng, G. A Synchronization Approach to Trajectory Tracking of Multiple Mobile Robots While Maintaining Time-Varying Formations. *IEEE Trans. Robot.* **2009**, *25*, 1074–1086. [[CrossRef](#)]
- Li, H.; Jiang, W.; Zou, D.; Yan, Y.; Zhang, A.; Chen, W. Robust motion control for multi-split transmission line four-wheel driven mobile operation robot in extreme power environment. *Ind. Robot. Int. J. Robot. Res. Appl.* **2020**, *ahead-of-print*. [[CrossRef](#)]
- Algrnaodi, M.; Saad, M.; Saad, M.; Fareh, R.; Kali, Y. Extended state observer-based improved non-singular fast terminal sliding mode for mobile manipulators. *Trans. Inst. Meas. Control* **2023**, *ahead-of-print*.

12. Shafei, A.M.; Mirzaeinejad, H. A novel recursive formulation for dynamic modeling and trajectory tracking control of multi-rigid-link robotic manipulators mounted on a mobile platform. *Proc. Inst. Mech. Eng. Part I J. Syst. Control Eng.* **2020**, *235*, 1204–1217. [[CrossRef](#)]
13. Portillo Vélez, R.d.J. Control Multilateral de Agarre para Robots Cooperativos Maestro/Multi-Esclavo. Ph.D. Thesis, CINVESTAV, Zacatenco, México, 2013.
14. Fierro, R.; Lewis, F. Control of a nonholonomic mobile robot: Backstepping kinematics into dynamics. In Proceedings of the 1995 34th IEEE Conference on Decision and Control, New Orleans, LA, USA, 13–15 December 1995; Volume 4, pp. 3805–3810. [[CrossRef](#)]
15. Bischoff, R.; Huggenberger, U.; Prassler, E. KUKA youBot—A mobile manipulator for research and education. In Proceedings of the 2011 IEEE International Conference on Robotics and Automation, Shanghai, China, 9–13 May 2011; pp. 1–4. [[CrossRef](#)]
16. Galati, R.; Mantriota, G. Path Following for an Omnidirectional Robot Using a Non-Linear Model Predictive Controller for Intelligent Warehouses. *Robotics* **2023**, *12*, 78. [[CrossRef](#)]
17. Wang, T.; Wu, Y.; Liang, J.; Han, C.; Chen, J.; Zhao, Q. Analysis and Experimental Kinematics of a Skid-Steering Wheeled Robot Based on a Laser Scanner Sensor. *Sensors* **2015**, *15*, 9681–9702. [[CrossRef](#)] [[PubMed](#)]
18. Fuentevilla, G. Definiciones del modelo KUKA youBot. 2022. Available online: [https://drive.google.com/file/d/1O\\_iH2lNSz0y-z2rv-nDqPhUTe0-oBxVb/view?usp=sharing](https://drive.google.com/file/d/1O_iH2lNSz0y-z2rv-nDqPhUTe0-oBxVb/view?usp=sharing) (accessed on 7 September 2023).
19. Gutiérrez, H.; Morales, A.; Nijmeijer, H. Synchronization Control for a Swarm of Unicycle Robots: Analysis of Different Controller Topologies. *Asian J. Control* **2017**, *19*, 1822–1833. [[CrossRef](#)]
20. Khalil, H.K. *Nonlinear Control*; Pearson Education: Upper Saddle River, NJ, USA, 2015.
21. Gómez León, B.C. Control de Manipulador Móvil Subactuado, Provisto de Unión Flexible. Master’s Thesis, CINVESTAV, Zacatenco, México, 2021.
22. Craig, J.J. *Introduction to Robotics: Mechanics and Control*; Pearson Education: Upper Saddle River, NJ, USA, 2005.
23. MediaWiki. Main Page—youBot Wiki. 2015. Available online: [http://www.youbot-store.com/wiki/index.php/Main\\_Page](http://www.youbot-store.com/wiki/index.php/Main_Page) (accessed on 7 September 2023).

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.